

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-305366
(43)Date of publication of application : 28.11.1997

(51)Int.Cl. G06F 3/14
G06F 3/14
G09G 5/14

(21) Application number : 08-117313 (7)

(71)Applicant : HITACHI LTD.

(22) Date of filing : 13.05.1996

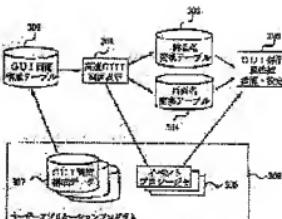
(72)Inventor : KUDO YUTAKA
YOSHIKAWA AKIHIRO

(54) SCREEN DISPLAY OPTIMIZATION METHOD

(57) Abstract:

PROBLEM TO BE SOLVED: To save the time required for an initialization processing and to display a screen at a high speed by resetting the attributes of a graphic user interface (abbreviated as GUI hereafter) screen initialized beforehand and GUI components at the time of eliminating a certain screen and displaying the other screen.

SOLUTION: In the case of executing a processing for performing setting or reference to the attribute value of the GUI component inside a display screen inside a user application program 308 and in the case of calling an event processing routine inside the user application program 308 from the display screen, a screen name and a component name are converted by a screen name conversion table 304 and a component name conversion table 303 and the respective processings are made executable. Also, the initialization processing of the GUI screen itself and the initialization processing of the GUI component for constituting the screen are performed beforehand, and at the time of eliminating the certain screen and displaying the other screen, the screen is reconstituted and displayed by changing screen contents without actually eliminating it.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application conversion or registration]

[Date of final disposal for application]

[Date of final dis]

[Entity number]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's rejection]

decision or rejection]

特開平9-305366

(13) 公開日 平成9年(1997)11月28日

(51) Int.Cl. ⁶	識別記号	序内整理番号	F I	技術表示箇所
G 06 F 3/14	3 6 0		G 06 F 3/14	3 6 0 C
	3 7 0			3 7 0 A
G 09 G 5/14			G 09 G 5/14	Z

審査請求 未請求 請求項の数2 O.L. (全10頁)

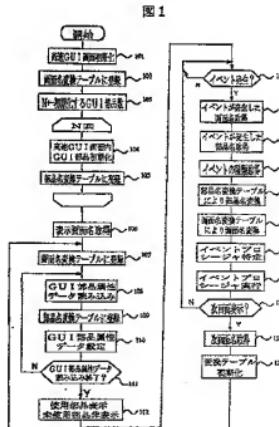
(21) 出願番号	特願平8-117313	(71) 出願人	000005108
(22) 出願日	平成8年(1996)5月18日	株式会社日立製作所	東京都千代田区神田駿河台四丁目6番地
		(72) 発明者	工藤 祐 神奈川県川崎市麻生区王禅寺1099番地株式会社日立製作所システム開発研究所内
		(72) 発明者	吉川 彰弘 神奈川県川崎市麻生区王禅寺1099番地株式会社日立製作所システム開発研究所内
		(74) 代理人	弁理士 小川 勝男

(54) [発明の名称] 画面表示最適化方法

(57) [要約]

【課題】異なる多數のグラフィカルユーザーインターフェース (GUI) 画面を有するユーザーアプリケーションプログラム (UAP)において、所望の GUI 画面を高速に表示すること。また、従来、メモリに保持しきれない程多数の GUI 画面をもつ UAP をメモリ容量の少ない小型の計算機で実行可能とすること。

【解決手段】グラフィカルユーザーインターフェース (GUI) 画面表示の度に発生する GUI 画面と GUI 部品の初期化処理を予め行っておき、GUI が画面表示の際にはそれらの属性値の変更のみで画面を再構成することによって可能とする。また、GUI 画面の構成情報を外部記憶装置に保持することによって可能とする。



【特許請求の範囲】

【請求項1】計算機及び対話端末を有し、該対話端末にグラフィカルユーザインターフェースを表示する画面制御プログラムと、ある画面の画面名と、該画面上に配置するグラフィカルユーザインターフェース部品の部品名と、座標と、大きさとの対応を示す画面構成データと、該画面構成データ中の該画面名を指定して該画面制御プログラムに画面表示を指示し、該画面構成データ中の該部品名を指定して該画面制御プログラムにグラフィカルユーザインターフェース部品に対する操作を指示する、アプリケーションプログラムとを有する情報処理装置において、該画面制御プログラムは、該画面構成データを記憶するステップと、画面表示の指示があったとき迅速に表示できるように予め初期化するステップと、該画面上に配置するグラフィカルユーザインターフェース部品の部品名と、他の部品と重複がないように固有の名前を割り当てるステップと、該画面上に複数のグラフィカルユーザインターフェース部品を記憶し、グラフィカルユーザインターフェース部品表示の指示があったとき迅速に表示できるように予め初期化するステップとを有し、該アプリケーションプログラムが画面表示を指示したときは、表示する画面の画面名を記憶するステップと、該画面名に対応するグラフィカルユーザインターフェース部品の部品名と、座標と、大きさを取得するステップと、前記ステップで取得した第1の部品名と、該画面上に配置したグラフィカルユーザインターフェース部品に前記ステップで固有の名前を割り当てる第2の部品名との対応を記憶するステップと、前記ステップで取得したグラフィカルユーザインターフェース部品の座標と、大きさに従って、該第2の部品名が示すグラフィカルユーザインターフェース部品の座標と大きさを変更するステップとによって画面表示を行ない、該アプリケーションプログラムがあるグラフィカルユーザインターフェース部品に対する操作を指示したときは、操作するグラフィカルユーザインターフェース部品の部品名を記憶するステップと、該部品名を該第1の部品名としたときに対応する該第2の部品名を取得するステップと、この該第2の部品名に対する操作を実行するステップとによってグラフィカルユーザインターフェース部品に対する操作を行なうことを特徴とする画面表示最適化方法。

【請求項2】請求項1記載の画面表示最適化方法において、前記該アプリケーションプログラムはさらに、表示した

画面上のグラフィカルユーザインターフェース部品に対して利用者が入力指示を行なったときに実行する処理プログラムと、該画面の画面名と該グラフィカルユーザインターフェース部品の部品名と入力指示の種類とに対応付けて保持し、該画面制御プログラムはさらに、

表示した画面上のグラフィカルユーザインターフェース部品に対して利用者が入力指示を行なったときは、入力指示を受けた該グラフィカルユーザインターフェース部品の部品名と入力指示の種類とを取得するステップと、該部品名を該第2の部品名としたときに対応する該第1の部品名を取得するステップと、この該第1の部品名と、該アプリケーションプログラムが画面表示を指示したときに前記ステップにおいて記憶した画面名と、該入力指示の種類とに対応づけて該アプリケーションプログラムが保持する該処理プログラムを実行するステップとによって利用者の入力指示に応じて処理を実行することを特徴とする画面表示最適化方法。

20 【明細の詳細な説明】

【0001】

【明細の属する技術分野】本発明は、計算機上のアプリケーションが有する異なる多数のグラフィカルユーザインターフェース画面の中から所望のグラフィカルユーザインターフェース画面をディスプレイ画面に高速に表示する方法に関する。特にグラフィカルユーザインターフェース部品を用いた容易な開発環境で構築されたアプリケーションをメモリ容量の少ない小型の計算機で実行し、そのアプリケーションが有する異なる多数のグラフィカルユーザインターフェース画面の中から所望のグラフィカルユーザインターフェース画面を高速に表示する場合に好適な画面表示最適化方法に関する。

【0002】

【従来の技術】グラフィカルユーザインターフェース画面を有するアプリケーションの構造には、開発の容易性から一般にグラフィカルユーザインターフェース部品と呼ばれるライブラリを用いる。

【0003】例えば、Microsoft社のVisual Basic 2.0のマニュアルp. 19~p. 20によれば、マウスを用いてグラフィカルユーザインターフェース部品を画面に配置することによってグラフィカルユーザインターフェース画面を作成できた。また、同マニュアルp. 21によれば、グラフィカルユーザインターフェース部品の属性値を設定することができた。また、同マニュアルp. 23によれば、イベント処理ルーチンを部品名とイベントの種類に対応付けて定義することができた。

【0004】また、異なる多数のグラフィカルユーザインターフェース画面を有するアプリケーションを作成し実行するために、あるグラフィカルユーザインターフェ

ース画面を非表示にする際にその画面のためのメモリ領域を解放し、他のグラフィカルユーザーインターフェース画面の表示の際にその画面のためのメモリ領域を動的に割り当てるという仕組みが提供された。

【0005】また、UNIX MAGAZINE 1994. 7 p. 67に掲載された「インターネットの利用と仕組み 1.5」によれば、サーバーからネットワークを介して送られた画面情報をクライアントで受け取り、クライアントがその画面情報を解釈して画面を表示するWWWの概念が開示された。

【0006】

【発明が解決しようとする課題】しかし、上記V! sua! Basicに見られる従来技術で作成されたユーザーインターフェース画面の数に比例してメモリの消費量が増加するため、表示可能な画面の数に限界があり、多数のGUI画面を有するユーザーインターフェースプログラムを実行することができなかった。

【0007】また、上記グラフィカルユーザーインターフェース画面を非表示にする際にその画面が使用していたメモリ領域を解放し他のグラフィカルユーザーインターフェース画面の表示の際にその画面に必要なメモリ領域を動的に割り当てるながら動作するという従来技術では、メモリ領域の解放及び画面を管理するための初期化処理に時間がかかるため、あるグラフィカルユーザーインターフェース画面を非表示にして他のグラフィカルユーザーインターフェース画面を表示する動作の速度が遅くなるという問題があった。

【0008】また、前記、WWWの概念に見られる従来技術では、クライアントは画面の表示をするだけで、アプリケーションプログラムを実行することなく、入力チェック処理等のアプリケーションに固有の処理を実行することができないという問題があった。

【0009】本発明は上記のような問題を解決するためになされたもので、その目的とするところは、従来技術における上記のような問題を解消し、グラフィカルユーザーインターフェース部品を用いた容易な開発環境で構築されたアプリケーションであっても使用メモリ量を抑え、かつ、高速にグラフィカルユーザーインターフェース画面を表示する画面表示最適化方法を提供することにある。

【0010】

【課題を解決するための手段】本発明では、画面構成情報を外部記憶装置に保持しておき、画面表示命令を実行した際に、これらの情報を実際にはメモリ中に1個しか存在しない画面に対して設定することによって所望の画面を構成し表示する。こうすることで、従来技術ではメモリに保持しきれないほどの多数のGUI画面を有するアプリケーションを実行可能とする。その際に、ユーザーインターフェースプログラム内で表示画面内のグラフ

カルユーザーインターフェース部品の属性値に対する設定または参照する処理を実行した場合と、表示画面からユーザーインターフェース部品の内部のイベント処理ルーチンを呼び出す場合には、画面名変換テーブルと部品名変換テーブルにより画面名と部品名を変換し、それぞれの処理を実行可能とする。

【0011】また、グラフィカルユーザーインターフェース画面自体の初期化処理とその画面を構成するグラフィカルユーザーインターフェース部品の初期化処理を予め行なっておき、ある画面を消去して他の画面を表示する際には、実際には消去せずに画面内容を変更することで画面を再構成し表示する。こうすることで、従来、画面の初期化に要していた時間を省くことができ、高速に所謂の画面を表示することを可能とする。

【0012】

【発明の実施の形態】以下、本発明の実施例について図面を参照して説明する。

【0013】図5は本実施例のハードウェア構成図である。図5において、505はCPUであり、グラフィカルユーザーインターフェース画面を表示するディスプレイ画面502と、磁気ディスクなどの外部記憶装置501と、メモリ503と、キーボードやマウス等の入力装置504がつながっている。

【0014】図3は本発明の実施例に係る高速画面表示最適化方法の処理機能とデータの関係を示すブロック図である。また、図4は従来の画面表示方法の処理機能とデータの関係を示すブロック図である。図3と図4において、長方形は処理機能を表し、円筒形はデータを表し、矢印はデータの流れを表す。

【0015】図3において、307はグラフィカルユーザーインターフェース(以後GUIと略す)画面構成データであり、開発時に作成されたGUI画面を構成するGUI部品の属性値を画面毎に保持している。306はイベントプロセッサーである。308はユーザーインターフェースプログラム(以下、UAPと略す)であり、307と306でひとつのUAPを構成する。GUI画面構成データ307とイベントプロセッサー306は、GUI部品を用いた容易な開発環境によって開発可能で、図4に示した従来の画面表示方法401で動作するUAP308と同一のものである。

【0016】302はGUI画面構成テーブルであり、GUI画面毎に存在するGUI画面構成データ307をテーブルの形式に変換して1つにまとめたものである。GUI画面構成データ302には、GUI画面名とそのGUI画面を構成するGUI部品の属性値がテーブルの形式で外部記憶装置501に保持されている。301は高速GUI画面表示機能であり、GUI画面構成データ302から表示する画面を構成するGUI部品の属性値を取得してGUI画面を作成し表示する。303は部品名変換テーブルであり、304は画面名変換テーブ

ルである。部品名変換テーブル303と画面名変換テーブル304は、GUI画面表示の際に高速GUI画面表示表301が作成し、メモリ503中に保持する。

【0017】305はGUI部品の属性値を参照・設定する機能であり、イベントプロセッサーを含むUAP308内の処理ルーチン内でGUI部品の属性値を参照・設定する命令が実行されたときに実行する。

【0018】ここで、本実施例の説明に用いるUAP308の例について図6を参照して説明する。図6はこのUAPが有するGUI画面の画面遷移を示した図である。このUAPは1個の業務メニュー画面601と、複数個の業務画面（業務1画面602と業務2画面603）を有し、ある画面を消して別の画面を表示することを繰り返しながら業務処理を進める対話型のUAPである。

【0019】図7は業務メニュー画面601の画面例であり、文字の表示を行なうテキスト型GUI部品701とユーザーのマウスクリック操作を受け付けるボタン型GUI部品（702と703）から構成される。図7において、ユーザーによってマウス等の入力装置504によりボタン型GUI部品702がクリックされると、業務メニュー画面601を消して業務1画面602を表示し、ボタン型GUI部品703がクリックされると業務メニュー画面601を消して業務2画面603を表示する。図8は業務1画面602の画面例であり、文字の表示を行なうテキスト型GUI部品801～803と、文字の入力を受け付けるテキスト型GUI部品804と805と、ユーザーのマウスクリック操作を受け付けるボタン型GUI部品806から構成される。ユーザーによってキーボード等の入力装置によりテキスト型GUI部品804と805にデータが入力され、次にマウスなどの入力装置504によってボタン型GUI部品806がクリックされると、氏名、生年月日の登録処理を行ない、業務1画面602を消して業務メニュー画面601を表示する。

【0020】ここで、GUI画面構成テーブル302について図9を参照して説明する。図9は、業務メニュー画面601と業務1画面602に対するGUI画面構成テーブル302の具体例である。図9において、901～909の各行はそれぞれ1個のGUI部品の属性データである。901～903は画面名「業務メニュー画面」、904～909は画面名「業務1画面」に対するGUI部品の属性データである。1個のGUI部品の属性データは、画面名910と、部品名911と、部品の種類を表す部品種別912と、部品の表示位置を表すx座標913とy座標914と、部品の大きさを表す幅915と高さ916と、部品上に表示する文字列を表すラベル917から構成される。

【0021】以下に図1と図2を参照して本発明の実施例の処理手順を、図6で示した画面遷移をするUAPを

例にして詳細に説明する。図1は図3における高速GUI画面表示301の処理手順を示すフローチャートであり、図2は図3におけるGUI部品属性値参照・設定処理305の処理手順を示すフローチャートである。

【0022】UAPからの要求により高速画面表示301がステップ101を実行すると、まず初めに、表示する画面の画面名を他の画面名と重複しない画面名、例えば「高速GUI画面」と決定し、高速GUI画面を初期化してディスプレイ画面502に表示する（ステップ101）。次に、初期化した高速GUI画面の画面名を画面名変換テーブル304に登録する（ステップ102）。この段階での画面名変換テーブル304の例を図10に示す。ここでは、高速GUI画面名の欄1001に「高速GUI画面」1003を登録し、UAP内で参照する画面名の欄1002を空欄にする。

【0023】次に、ステップ101で初期化された高速GUI画面に表示するGUI部品の最大の個数を変数Nに代入し（ステップ103）、N個分のGUI部品を初期化（ステップ104）、初期化した部品名を部品名変換テーブル303に登録する（ステップ105）。初期化するGUI部品の個数は計算機のメモリ容量やオペレーティングシステムにより異なるのであからかじめ快適に動作する最大数のGUI部品数を調べておく必要がある。この例では、テキスト型GUI部品5個とボタン型GUI部品5個を初期化するものとする。ステップ104で初期化したテキスト型GUI部品の部品名を「Text1」、「Text2」、…、「Text5」、ボタン型GUI部品の部品名を「Button1」、「Button2」、「…」、「Button5」とし、このときの部品名変換後の部品名変換テーブル303の例を図11に示す。ここでは、高速GUI画面内部品名の欄1101に「Text1」、「Text2」、…、「Text5」、「Button1」、「Button2」、「…」、「Button5」1103を登録し、UAP内で参照する部品名の欄1102は空欄にする。

【0024】ステップ101～105は、画面表示の指示があったときに迅速に表示できるように予め初期化処理をするステップである。

【0025】次に、UAPが最初に表示する画面の画面名を取得し（ステップ106）、画面名を画面名変換テーブル304に登録する（ステップ107）。図6の例に従えば、最初に表示する業務メニュー画面601の画面名である「業務メニュー画面」を画面名変換テーブル304に登録する。登録後の画面名変換テーブル304の例を図12に示す。図12において、ステップ102で登録した「高速GUI画面」1203に対応付けて、UAP内で参照する画面名「業務メニュー画面」1201を登録する。

【0026】次に、ステップ106で得た画面名をキーにしてGUI画面構成テーブル302からGUI部品属

性データを検索し、この画面名を構成する GUI 部品の属性値を取得する（ステップ 108）。例えば、画面名「業務メニュー画面」をキーにして図 9 に示した GUI 画面構成テーブルから検索し、GUI 部品属性データ 901 を取得する。

【0027】次に、ステップ 108 で取得した GUI 部品属性データ中の部品名を部品名変換テーブル 303 に登録する（ステップ 109）。GUI 部品属性データ 901 によれば、部品名「Title」を部品名変換テーブル 303 に登録する。部品名変換テーブルの例を図 13 に示す。図 13 の 1102 はステップ 109 が部品名を登録する欄で、この段階では 1301 に部品名「Title」を登録する。次に、GUI 部品属性データ 901 の各値をステップ 104 で初期化した GUI 部品の属性値に設定する（ステップ 110）。GUI 部品の属性値に GUI 部品属性データ 901 を設定することによって、GUI 部品属性データ 901 が示す、x 座標が 10、y 座標が 1 の位置に、幅が 15、高さが 1 の大きさで、「業務メニュー」と表示した GUI 部品を配置する。

【0028】ステップ 111 は表示する画面名に対する GUI 部品データ取得が終了したかどうかを判定するステップで、画面名「業務メニュー画面」に対しては図 9 における 901～903 についてステップ 108～ステップ 110 を繰り返す。

【0029】901～903 についてステップ 108～110 が完了したときの部品名変換テーブル 303 の例を図 13 に示す。図 13において、1103 はステップ 105 で登録された高速 GUI 画面内の部品名である。ステップ 109 は、GUI 部品属性データ 901 に対しては「Title」1301を、GUI 部品属性データ 902 に対しては「Gyomu1」1302を、GUI 部品属性データ 903 に対しては「Gyomu1」1303をそれぞれ登録する。

【0030】次に、部品名変換テーブル 303 に登録した GUI 部品を表示状態にし、登録していない GUI 部品を非表示状態にする（ステップ 112）。図 13 に示した部品名変換テーブルにおいて、UAP 内で参照する部品名の欄 1102 が空欄のもの 1304 と 1305 が未使用の GUI 部品である。

【0031】ステップ 106～112 は、ステップ 101～105 で初期化した GUI 画面と GUI 部品に、ステップ 106 で取得した画面名に対する GUI 画面構成データを反映させて実際に画面を表示するステップである。

【0032】ステップ 101～112 によれば、従来、実行時にメモリ中に保持していないければならなかった GUI 画面構成データを、外部記憶装置に保持し、実行時に画面毎に読み込みながら GUI 画面を表示することが可能となる。GUI 画面構成データを外部記憶装置に保

持し、実行時に画面毎に読み込みながら GUI 画面を表示することによって、メモリ容量の少ない小型の計算機でも、異なる多款の画面を有する UAP の実行が可能となる。

【0033】ここで、イベントについて説明する。イベントは GUI 画面上の GUI 部品に対してユーザーからのマウスクリックやキー押下などによって発生する。また、イベントプロセッサーとは、イベントが発生した GUI 部品とイベントの種類に対応した処理ステップのことである。

【0034】ステップ 113 はイベントが発生したかどうかを検査するステップであり、イベントが発生しない限り次のステップ 114 へは進まない。表示されている画面に対してユーザー入力等によってイベントが発生した時は、イベントが発生した画面名と部品名をそれぞれ取得し（ステップ 114、ステップ 115）、発生したイベントの種類を取得する（ステップ 116）。ステップ 113 で取得した画面名は高速 GUI 画面名であり、ステップ 115 で取得した部品名は高速 GUI 画面内の部品名である。そこで、UAP 内のイベントプロセッサーを呼び出すために、部品名変換テーブル 303 と画面名変換テーブル 304 を用いて UAP 内で参照する部品名と画面名にそれぞれ変換する（ステップ 117 とステップ 118）。ステップ 117 で取得した部品名と、ステップ 118 で取得した画面名と、ステップ 116 で取得したイベントの種類から UAP 内のイベントプロセッサーを特定し（ステップ 119）、実行する（ステップ 120）。

【0035】図 6 の例に従えば、業務メニュー画面 601 に表示されたボタン型 GUI 部品 702 がユーザーによってマウス等の入力装置でクリックされると、高速 GUI 画面に表示されたボタン型 GUI 部品 702 で「クリック」というイベントが発生し、ステップ 114 によって画面名「高速 GUI 画面」、ステップ 115 によって部品名「Button1」、ステップ 116 によってイベントの種類「クリック」を取得する。これらの画面名と部品名を画面名変換テーブル 304 と部品名変換テーブル 303 によって変換し、UAP で参照する画面名「業務メニュー画面」と部品名「Gyomu1」を取得する（ステップ 118、117）。この画面名「業務メニュー画面」と、部品名「Gyomu1」と、「クリック」というイベントの種類によって UAP 内のイベントプロセッサーを特定し、実行する。

【0036】ステップ 114～120 によれば、画面名変換テーブル 304 と部品名変換テーブル 303 を用いることによって、実際にメモリ中に存在する GUI 画面の画面名と部品名が UAP 内の画面名と部品名と異っていても、UAP 内のイベントプロセッサーを実行することができる。この点が本発明の新規な点のひとつである。

【0037】ステップ120で実行するイベントプロセッサ内での画面を切り替える命令を実行する時(ステップ121)は、先ず、次に表示する画面名を取得する(ステップ122)。次に、部品名変換テーブル303のUAP内で参照する部品名の欄1102と、画面名変換テーブル304のUAP内で参照する画面名1002の欄を空欄にし(ステップ123)、ステップ107に戻り、画面表示のためのステップ107~110を繰り返す。

【0038】ある画面を消去して他の画面を表示する際に、実際に消去せずに画面内容を変更することで画面を再構成し表示することによって、従来必要だったGUI画面の初期化(ステップ101)とGUI部品の初期化(ステップ104)にかかる時間を省いた点が本発明の新規な点である。

【0039】図6の例に従って、業務メニュー画面601を消して、次に業務1画面602を表示する処理手順を説明する。

【0040】先ず、イベントプロセッサ内での画面切り替え命令を実行すると、ステップ122で次に表示する画面名「業務1画面」を取得する。そしてステップ123により、部品名変換テーブル303と画面名変換テーブル304を図10、図11の状態に初期化する(ステップ123)。以下、ステップ107~112を繰り返す。

【0041】画面名「業務1画面」に対してステップ107~110を実行した後の画面名変換テーブル304を図14に、部品名変換テーブル303を図15に示す。ステップ107は図14におけるUAP内で参照する画面名の欄1401に「業務1画面」を登録し、ステップ109は図15におけるUAP内で参照する部品名の欄1102に1501を登録する。

【0042】イベントプロセッサを含むUAP内の処理ルーチン内で、画面に記されている部品の属性値を参照もしくは設定する命令文が存在した場合の処理手順を図2を参照して説明する。先ず、参照もしくは設定の対象となるGUI部品の名前を命令文から取得し(ステップ201)、部品名変換テーブル303を用いて高速GUI画面内の部品名に変換する(ステップ202)。同様に画面名も画面名変換テーブル304を用いて高速GUI画面名に変換する。次に、行う処理が部品属性値の参照処理か設定処理かを判断し(ステップ204)、参照処理ならばステップ203で得た画面名が指示する高速GUI画面内のステップ202で得た部品名が指示するGUI部品の属性値を参照する(ステップ205)。設定処理の場合も同様にステップ203で得た画面名が指示する高速GUI画面内のステップ202で得た部品名が指示するGUI部品の属性値を参照する(ステップ206)。

【0043】以上に説明した処理手順によれば、GUI

画面とGUI画面上に表示するGUI部品の初期化を一度だけ行い、以後、ある画面を消去して他の画面を表示する際には、予め初期化されたGUI画面、GUI部品の属性を設定しなおすことにより、画面構成を変更し、所望のGUI画面を得ることができる。このため、従来、ある画面を消去して他の画面を表示する度に発生していたGUI画面およびGUI画面上に表示するGUI部品の初期化処理にかかる時間を省くことができ、高速に画面を表示することができる。

【0044】また、UAPが有する多数の画面構成情報を外部記憶装置501に保持することで、計算機のメモリ503内には1個分のGUI画面のためのメモリ領域しか必要とせず、メモリ容量の少ない小型の計算機でも事なる多数の画面を有するUAPを実行することができる。

【0045】さらに、部品名変換テーブル303、画面名変換テーブル304を用いることにより、従来通りのGUI部品を用いた容易な開発環境で開発されたUAPであってもソースプログラムの変更無くそのまま実行することができる。

【0046】

【発明の効果】本発明によれば、GUI部品を用いた容易な開発環境で開発されたUAPであっても、ソースプログラムの変更無く、異なる多数の画面を持つアプリケーションプログラムをメモリ容量の少ない小型の計算機で実行することができる。しかも、高速な画面表示が可能となる。

【0047】例えば、従来、ある画面を消去して他の画面を表示する度に発生していたGUI画面およびGUI画面上に表示するGUI部品の初期化を行っていたが、これらの初期化処理を予め行っておき、以後、画面表示の際には、外部記憶装置から得たGUI部品属性のみを設定しなおすことにより、高速な画面表示が可能となる。

【0048】また、UAPが有する多数の画面構成情報を外部記憶装置に保持することで、計算機のメモリ内には1個分のGUI画面のためのメモリ領域しか必要とせず、メモリ容量の少ない小型の計算機でも多数の画面を有するUAPを実行することができる。

【0049】さらに、部品名変換テーブル、画面名変換テーブルを用いることにより、従来通りのGUI部品を用いた容易な開発環境で開発されたUAPであってもソースプログラムの変更無くそのまま実行することができる。

【画面の簡単な説明】

【図1】本発明に係る高速画面表示最適化方法の処理手順を示すフローチャートである。

【図2】ステップ102において、GUI部品属性値を設定もしくは参照する処理手順を示すフローチャートである。

【図3】本発明に係る高速画面表示最適化方法の処理機能とデータの関係を示すブロック図である。

【図4】従来の画面表示方法の処理機能とデータの関係を示すブロック図である。

【図5】本発明に係る高速画面表示最適化方法のハードウェア構成図である。

【図6】本発明の対象となるアプリケーションプログラムが有するグラフィカルユーザーインターフェース画面の画面遷移を示す図である。

【図7】図6の業務メニュー画面601の画面例である。

【図8】図6の業務1画面602の画面例である。

【図9】GUI画面構成テーブル302の一例である。

【図10】ステップ102で生成する画面名変換テーブル304の一例である。

【図11】ステップ105で生成する画面名変換テーブ

ル303の一例である。

【図12】ステップ107で生成する画面名変換テーブル304の一例である。

【図13】ステップ110で生成する画面名変換テーブル303の一例である。

【図14】ステップ107で生成する画面名変換テーブル304の一例である。

【図15】ステップ110で生成する画面名変換テーブル303の一例である。

10 【符号の説明】

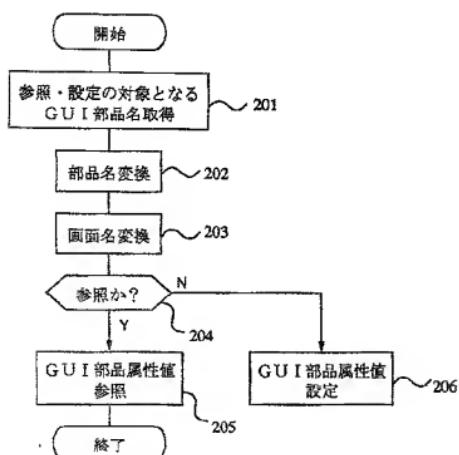
301…高速GUI画面表示、 302…GUI

画面構成テーブル、 303…部品名変換テーブル、

304…画面名変換テーブル、 305…GUI部品属性値参照・設定、 306…イベントプロシージャ、 308…ユーザーアプリケーションプログラム。

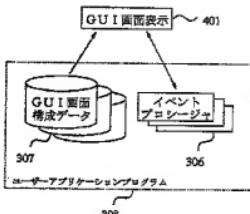
【図2】

図2



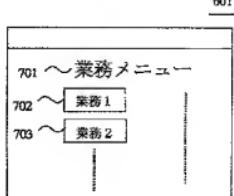
【図4】

図4



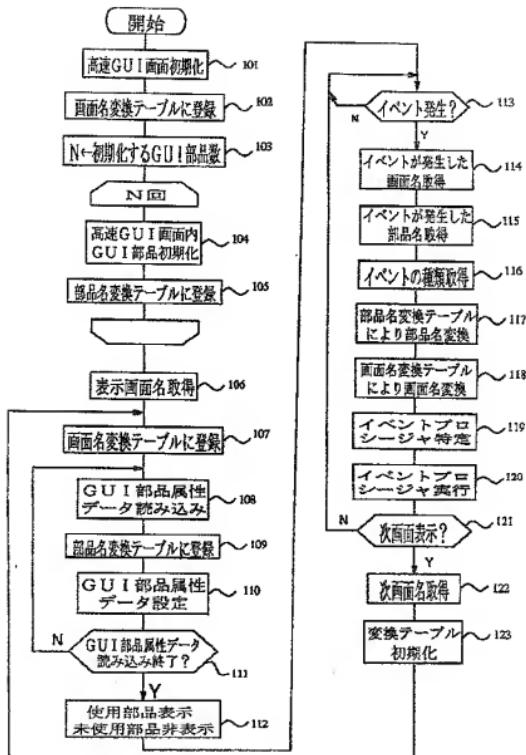
【図7】

図7



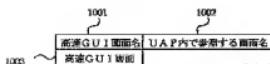
【図1】

図1



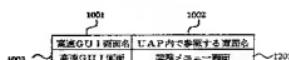
【図10】

図10



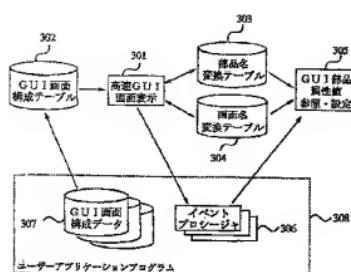
【図12】

図12



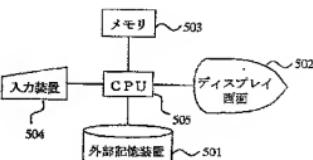
【図3】

図3



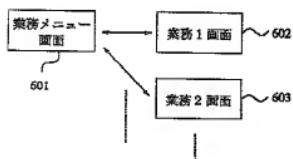
【図5】

図5



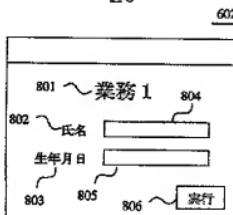
【図6】

図6



【図8】

図8



【図9】

図9

	910	911	912	913	914	915	916	917
画面名	部品名	商品別	1次部	2次部	新	既	ラベル	
901 ~ 業務メニュー画面	Title	Text	3	1	1	1	既選択	
902 ~ 業務メニュー画面	Gyomu1	Button	9	3	1	1	既選択	
903 ~ 業務メニュー画面	Gyomu2	Button	9	4	1	1	既選択	
904 ~ 業務1画面	Title	Text	11	1	1	1	既選択	
905 ~ 業務1画面	Label1	Text	7	3	1	1	既選択	
906 ~ 業務1画面	Label2	Text	7	4	1	1	既選択	
907 ~ 業務1画面	Name	Text	13	3	1	1	既選択	
908 ~ 業務1画面	Birthday	Text	13	4	1	1	既選択	
909 ~ 業務1画面	Enter	Button	20	10	5	1	既選択	

【図11】

図11

1101	1102
高級GUI画面内の画面名	UAPで利用する部品名
Text	
Text2	
Text3	
Text4	
Text5	
Button1	
Button2	
Button3	
Button4	
Buttons	

【図1.3】

図1.3

高速GUI画面内の項目名	UAPで参照する項目名
Text1	Title
Text2	
Text3	
Text4	
Text5	
Text6	
Button1	Label1
Button2	Label2
Button3	
Button4	Name
Button5	Birthday
Button6	Enter

【図1.4】

図1.4



【図1.5】

図1.5

高速GUI画面内の項目名	UAPで参照する項目名
Text1	Title
Text2	Label1
Text3	Label2
Text4	Name
Text5	Birthday
Text6	Enter
Button1	
Button2	
Button3	
Button4	
Button5	